基于 Prometheus+Grafana 实现新华全媒新闻服务平台统一运维监控

钟盈炯
（新华社通信技术局，北京 100803）

摘 要：【目的】为实现新华社供稿平台的统一运维监控，基于 Docker 搭建的新华全媒新闻服务平台（为新华社供稿平台，以下简称全媒平台），已成为新华社海内外供稿用户的收稿平台，对平台运行情况的监控是事关全媒平台稳定安全运行的重要保障之一。【方法】文章主要介绍基于使用开源工具 Prometheus 和 Grafana，实现基于 Docker 搭建全媒平台的系统监控方式方法。【结果】本文首先介绍了上述两种开源工具和 Docker 微服务部署的技术要点，再阐述了监控系统的技术架构，从而说明使用这两种工具搭建线上企业级运维监控系统的策略、实施方案和实施过程，实现界面展示和告警通知的整体监控效果。【结论】通过基于 Prometheus 和 Grafana 等开源框架，搭建企业级新华全媒新闻服务平台统一运维监控平台，验证了设计方案的可行性，提升了运维值班同志的工作效率，保证了系统的稳定性。

关键词：Docker；Prometheus；Grafana；微服务；新华全媒新闻服务平台 **中图分类号：**TN948.6 **文献标识码：**A
文章编号：1671-0134（2023）01-154-05 **DOI：**10.19483/j.cnki.11-4653/n.2023.01.031
本文著录格式：钟盈炯 . 基于 Prometheus+Grafana 实现新华全媒新闻服务平台统一运维监控 [J]. 中国传媒科技，2023（01）：154-158.

导语

新华全媒新闻服务平台是按照新华社供稿线路优化调整的总体要求，基于云计算和微服务技术架构，重构新华社转型发展时期的供稿技术体系，建成包括所有新华社文字、图片、图表、视频、新媒体、多媒体、历史资料的全媒体供稿平台。Docker 容器技术将作为云计算领域的代表技术之一，以镜像方式交付，以容器方式运行，使得软件所依赖的环境与标准环境相同，只需要进行一次构建，即可实现重复部署。用微服务来重新定义架构体系已成为业内系统设计和技术实现的热门方向和首要选择。

本文首先对 Docker、Prometheus 和 Grafana 进行了介绍，然后说明了基于 Docker 部署的微服务架构全媒平台，提出了其采用的 Prometheus+Grafana 实现企业级系统监控的设计方案和实现过程，最后展示了部分可视化输出效果。

1.Docker 简介

Docker 是一个被广泛使用的开源容器引擎，是一种操作系统级别的虚拟化技术，它以一种特殊进程的方式运行于宿主机上，它依赖于 liunx 内核特性：namespace（名字空间进行资源的隔离）和 cgroups（限制、记录任务组所使用的物理资源），它也可以对应用程序进行打包。Docker 是一种基于 LXC 的轻量级虚拟化技术，基于 Go 语言开发，并遵循 Apache2.0 协议。[9] 其主要目标是“Bulid，Shop and Run Any App，Anywhere”。即利用 Docker 容器的特点，对资源进行分割和调度，主要面向于开发者与系统管理员，最终实现一个分布式平台，主要负责管理应用组件的整个生命周期。使用 Docker 容器技术，可以对应用进行高效、敏捷且自动化的部署，同时结合操作系统内核技术（namespaces，cgroups 等），为 Docker 容器的安全与资源隔离提供了技术保障。[1]

2. Prometheus 与 Grafana 概述

Prometheus 是由 SoundCloud 开发的开源监控报警系统和时序列数据库（TSDB）。[2]

Prometheus 使用 Go 语言开发，是 Google BorgMon 监控系统的开源版本。2016 年由 Google 发起 Linux 基金会旗下的原生云基金会（Cloud Native Computing Foundation），将 Prometheus 纳入其下第二大开源项目。Prometheus 目前在开源社区相当活跃。Prometheus 和 Heapster（Heapster）是 K8S 的一个子项目，用于获取集群的性能数据。相比功能更完善、更全面。Prometheus 性能也足够支撑上万台规模的集群部署。[2]

Grafana 是开源的、炫酷的可视化监控、分析利器[3]，拥有快速灵活的客户端图表和模块工具，面板插件有许多不同方式的可视化指标和日志，官方库中具有丰富的仪表盘插件，比如热图、折线图、图表等多种展示方式，让复杂的数据展示得美观而优雅。支持许多不同的时间序列数据库作为其数据来源的源头，诸如本文中提及的 Prometheus。

3. Prometheus+Grafana 设计实现企业级系统运维监控

运维监控系统的实现过程是，将基础平台和业务系统中所涉及的硬件资源信息、基础组件信息、应用软件信息等统一纳入运维监控平台，并进行指标的规范、收集及统一集中存储。以可用性指标为基础，逐步增加服务质量相关指标。实现系统运维监控的规范化和故障告警处理的智能化。

运行监控和故障告警是运维监控系统的两个主要功能组成部分。根据上述实现思路，统一运维监控平台的实现架构设计如图 1 所示，划分为三大组成部分，分别是数据采集、数据提取（存储）和数据展示及报警提示。数据采集模块主要是部署 Exporter 等监控工具，获取各类基础数据，当针对具体的应用实现时，运维人员需要编写代码获取应用的监控指标，并格式化为 Prometheus 的数据格式形式；数据提取（储存）主要是将指标数据存储到 Prometheus 时序数据库中，主要用来存储和查询监控的指标数据；数据展示及报警提示模式主要是通过运用 Grafana 以及邮件、微信等外围输出工具，实现基础环境和业务系统监控指标的可视化展示和告警信息的输出。

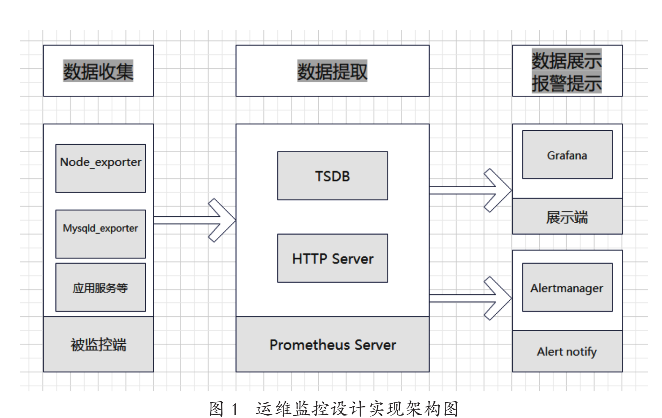


图 1 运维监控设计实现架构图

4. 基于容器微服务架构部署的全媒平台系统

4.1 微服务简介

微服务架构是一种面向互联网应用服务的软件开发架构[4]，主要应用于互联网应用服务的服务端软件开发，由面向服务架构 SOA 发展而来。微服务架构提倡将单体架构应用划分成一组小的服务，服务之间互相协调、互相配合。

4.2 从传统向微服务开发框架转移

笔者所在单位的原有全媒平台是基于开源的 dubbo 框架设计搭建而成，庞大而复杂，此架构对敏捷开发和迭代优化部署较为繁琐，尤其是在迭代升级和版本回退时较为困难。

本文中提及的现有全媒平台，由传统的服务架构向基于 Spring Cloud 的微服务架构转移，通过调用本地 Consul 客户端 Consul 服务器注册、发现和消费。向 Consul 服务器注册时，发告知其 IP 和端口，注册后，会每隔一定时间发送健康检查，当需要消费时，先去 Consul 服务器上拿到一个含有 IP 和端口的临时表，再去 Get 实际的路由。

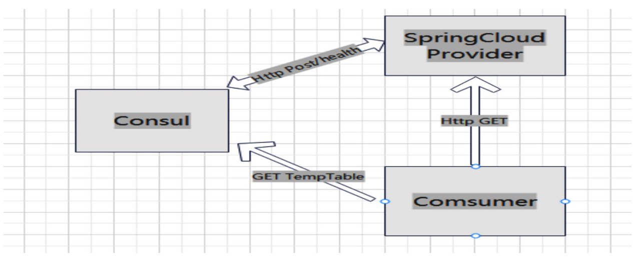


图 2 服务发现方式

4.3 业务服务部署架构

为实现全球站点部署，在设计业务的服务架构中充分考虑到业务的增减和变更情况。具体有：gateway（应用服务路由网关）、doc-view（稿件查看）、doc-server（稿件服务）、auth（用户认证授权）、management（后台资源管理）、consul server（服务注册与发现）和稿件入库等。其业务部署架构图如下所示。

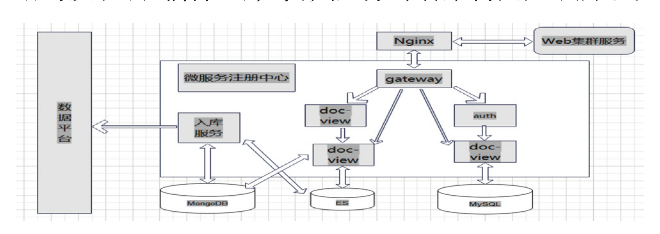


图 3 业务服务部署架构图

4.4 容器化部署实施

在上述对各个微服务模块进行细分的前提下，实现服务模块化打包、持续集成持续交付（CI/CD）的自动化运维服务能力。在此，笔者项目组使用 Docker，方便地以“容器化”的方式去部署应用服务，它在镜像中打包了所有应用所需要的环境，正所谓一次构建，处处运行。为了方便对 Docker 容器进行规模化和集群化管理，谷歌公司推出的 Kubernetes（简称 K8s）的容

器集群管理系统。Kubernetes 主要包括容器集群的自动化部署、自动扩缩容、容器维护管理等功能模块。[5] 在该项目中，使用 K8s 对各个应用 Docker 容器进行统一的管理，根据业务所需和访问情况动态扩充，以保证系统服务的稳定性、安全性和可靠性。

5. 搭建服务于全媒平台的企业级运维监控平台

5.1 功能架构

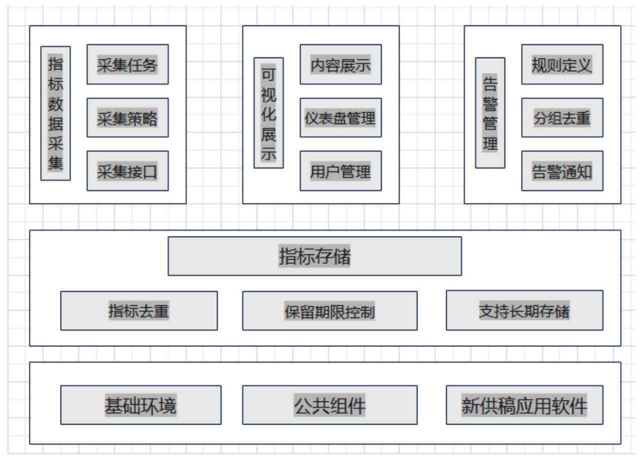


图 4 统一运维监控平台功能架构图

平台主要分为指标数据采集、指标存储、可视化展现、告警管理 4 个主要功能模块。指标采集模块负责对所有的指标接口进行数据采集，并将采集到的时序指标数据存入指标存储时序数据库中进行长期存储，可视化展现模块利用这些时序数据进行指标的各种展现形式的可视化呈现，告警管理模块则根据告警规则，结合时序数据进行规则匹配，若触发规则，则在分组去重后进行告警通知。

5.2 技术架构及实现

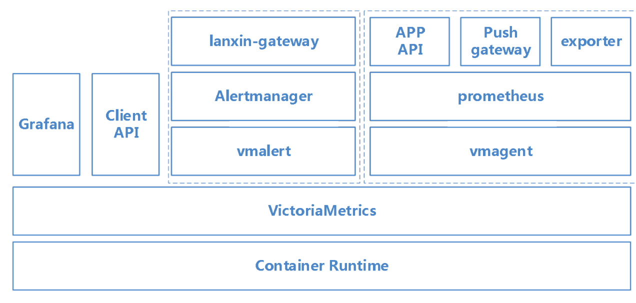


图 5 运维监控设计实现架构图

平台依托于开源技术、产品与自主研发的相关模块构建。底层依托容器环境运行。

主要开源技术、产品的选型介绍如下：

Prometheus（详情请见本文 3 中简述）。

在本系统中作为二级抓取服务，由于具有 Web UI 服务，所以更方便各二级抓取负责人员进行调试、更新、维护。

Grafana（详情请见本文 3 中简述）。

查询分析时序数据库中的时序数据然后进行可视化展示。

VictoriaMetrics

VictoriaMetrics 套件中的指标集中存储组件，是一个支持高可用、消耗低、可伸缩的时序数据库，用于 Prometheus 标准时序指标数据的长期存储。

vmagent

VictoriaMetrics 套件中的指标采集组件，可以比 Prometheus 更高效、资源占用更低的采集海量时序指标数据。

vmalert

VictoriaMetrics 套件中的告警指示组件，其执行一系列给定的 rule（基于 MetricsQL，PromQL 的超集），然后发送告警信息到 Alertmanager 组件。

Alertmanager

告警通知组件。其接收 vmalert 发送的告警信息，并通过各种告警通知渠道发送告警信息。可以做到告警信息进行去重，降噪，分组，策略路由。

lanxin–gateway

蓝信消息网关组件。接收 Alertmanager 发送的告警信息，进行预处理和格式转换后通过调用蓝信群消息接口将告警消息发往相应蓝信告警群。

5.3 部署架构

图 6 为运维监控平台部署架构示意图，计划在全球四大供稿站点和北京总社部署一台或者多台服务器，用于部署相应监控组件的服务。具体从功能上分为在 4 个应用服务站点部署二级抓取服务器，收集本站点的监控信息，统一收集后，发送至总社统一汇总统一管理；在北京总社部署集中指标收集服务器和核心服务器。

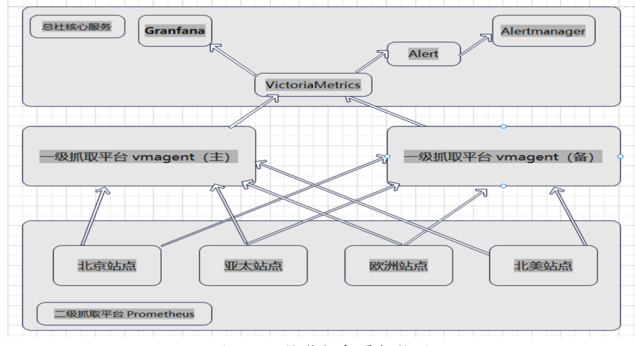


图 6 运维监控部署架构图

5.3.1 核心服务层

图 6 从上向下看，最上一层为核心服务层。主要

负责时序数据的存储，同时有告警规则运算服务 alert 负责告警规则的匹配运算，告警服务 Alertmanager 负责告警消息分组去重及分发，可视化展示服务 Grafana 提供指标展示。具体就是接收数据收集层发来的格式化规范化的数据，进行分析筛选，统一存储至时序数据库中，Grafana 调用时序数据库中的各类数据，选用合适的展示模板供用户查看，同时根据业务需要和业务要求编写各类告警规则，输出报警数据。

图 6 中间一级抓取层和最下面二级抓取层主要是收集服务器主机的基础数据、基础组件数据和所监控应用的服务数据等，将所收集到的数据进行标准化，提供给 Prometheus 的服务采集接口。具体为：

一级抓取层。负责拉取对应区域所有二级抓取机中抓取到的指标数据，并将这些数据存储到核心服务器的时序数据库中。一级抓取会将指标数据同时推送到总社的核心服务器上进行存储，保证数据可用性。

二级抓取层，负责抓取所管辖服务站点内的指标数据。具体抓取的数据有：抓取所有站点物理机和虚拟机的系统数据及基础组件的指标数据；全媒平台所涉及的 4 个站点的重要应用服务接口的指标数据，所获取的指标数据需足以保证足以覆盖业务服务状态的安全、可靠和稳定。

5.4 环境搭建实现过程

5.4.1 数据收集配置过程

在 4 个站点和总社汇总点，搭建统一运维监控集群服务。分为总社汇总、一级抓取和二级抓取。

在各个站点安装 exporter，实现基础数据的采集。主要指标数据分为 CPU、内存、文件系统、磁盘、网络、TCP 连接数等方面。包括 CPU 各模式秒数、5 分钟平均负载、内存总 / 空闲 / 可用字节数、文件系统总 / 可用字节数、mysql、mongoDB、Nginx、es、redis 等多项关键指标。

在各个站点部署应用的指标采集程序，安装白盒黑盒探针及日志分析服务等，转化为 Prometheus 可以识别可以支持的数据格式，转换为上一级可以提取的数据格式。

5.4.2 数据展示配置过程

登录总社服务器，安装 Grafana。（如果查看各个站点收集数据的展示情况，也可在分站点安装 Grafana）。

通过 Web 服务连接 Grafana，使用管理员账户登录 Grafana，配置连接的时序数据库数据源。

选用合适的展示模块，如当前没有，从官网获取相应的 Json 文件或者模块编号，将其导入 Grafana 中。

综合分析业务展示方式和展示效果，选取所需要的各类数据，包括基础数据和应用服务指标数据，将其展示到 Grafana 的 Web 界面中。

5.4.3 告警规则配置过程

Grafana 的告警触发以 panel 为基础，即每个 panel 单独配置告警信息，包括告警规则、触发条件、告警通知通道及内容。

指定所需要修配配置的通道，修改 Grafana 配置文件（grafana.ini）。

登录 Grafana Web 服务界面，进入设置区，接收告警的通道，并配置相应的阈值。

6. 整体界面效果

根据配置的 Granfa 服务地址和端口，登录 Grafana Web 服务，配置连接对应的时序数据库，将收集到的格式化数据统一展示在运维监控大屏之中，巧妙选择不同的显示颜色和不同的展示方式，将数据平面化、图形化、易读化，便于运维值班人员随时查看了解系统赖以运行的基础环境和网络的实时工作情况，第一时间获取系统运行的状态信息和报警信息。

6.1 全媒平台监控概览

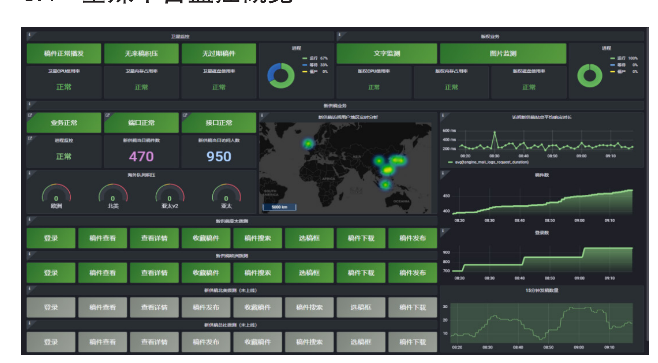


图 7　全媒平台统一监控总览

图 7 展示了全媒平台业务、端口、接口和进程的总体情况。如某个模块颜色变红，则说明存在报警情况，将鼠标放置在某一面板左上的超链接图标上，即显示下钻详情的超链接，点击可进入相应的二级监控页查看详情。左下部分为新供稿 2.0 四个站点的拨测详情。如某个模块颜色变红，则说明存在报警情况，点击相应模块可进入二级监控页查看详情。

6.2 全媒平台二级监控细览

图 8 和图 9 展示了业务系统中涉及的进程服务状态和端口服务状态，图 10 和图 11 展示了业务系统中部署的基础环境和网络环境的整体情况，下图的颜色会变成黄色或者红色等不同颜色状态信息，以方便运

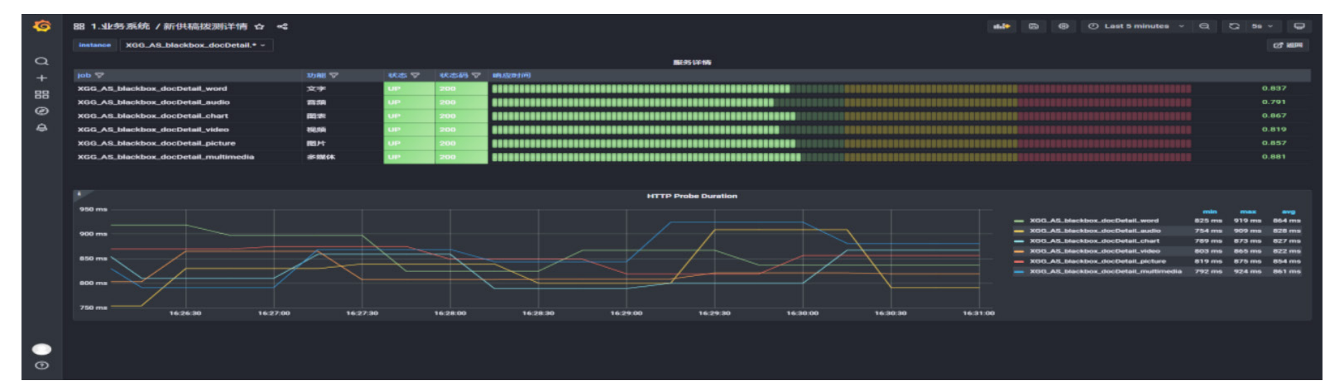


图 8　全媒平台二级拨测详情

维人员通过颜色及时获知系统的运行情况。

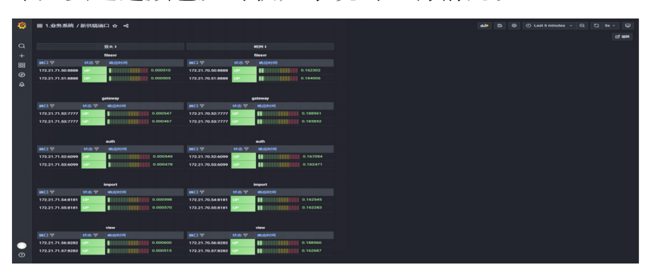


图 9　全媒平台二级端口服务详情



图 10　全媒平台 CPU/ 内存情况

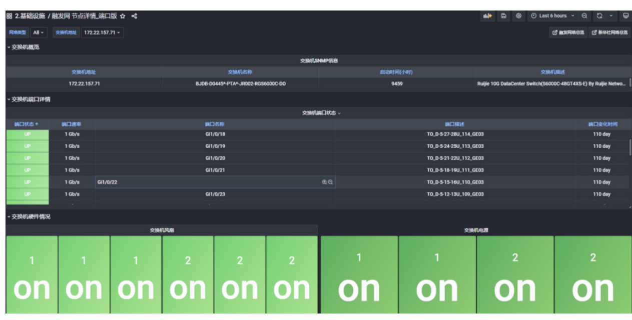


图 11　全媒平台网络基础环境情况

结语

简单而又直观的监控信息展示，是系统运维的利器，正确而又及时地监控报警是服务稳定的基础。随着微服务架构设计理念应用到全媒平台的实际生产应用中，开源的 Prometheus+Grafana 天然组合，因其简单、稳定、可靠和易扩展等特点，成为搭建企业级运维监控平台的首选环节。笔者在本文中所阐述的设计方案和实施细节，有效助力了全媒平台运维人员的运维效率和运维能力，从而进一步保障了系统的稳定可靠运行，成为当前系统运维不可或缺的组成部分。

参考文献

[1] 微畅享 .docker 简介 [EB/OL].https：//baijiahao.baidu.com/s?id=1692361731135557712&wfr=spider&for=pc.2021-02-22/2022-12-23.

[2] linux 人 .Prometheus 入门：开源监控报警系统和时序列数据库 [EB/OL].https：//ywnz.com/linuxysjk/2287.html.2018-07-14/2022-12-21.

[3] 可视化工具 Grafana 的简介 [EB/OL].http：//www.360doc.com/content/19/0711/19/16619343_848115684.shtml.2019-07-11/2022-12-22.

[4] 韩笑，李洁原 . 基于微服务架构的新闻制作系统优化探索 [J]. 中国传媒科技，2021（2）：62-63.

[5] 孙波 . 浅谈微服务架构、Docker 和 Kubernetes[J]. 现代电视技术，2022（2）：100-103.

[6] 邵珂，蔡国华，万国雷 . 中国搜索 Kubernetes 应用平台部署方案 [J]. 中国传媒科技，2019（5）：113-117.

作者简介：钟盈炯（1981-），男，浙江诸暨，硕士研究生，新华通讯社通信技术局，高级工程师。

（责任编辑：张晓婧）